Un nouveau réseau d'interconnexion adapté aux calculateurs SIMD

André Seznec

IRIA

CENTRE DE RENNES
IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tel. (3) 954 90 20

Rapports de Recherche

N° 329

UN NOUVEAU RÉSEAU D'INTERCONNEXION ADAPTÉ AUX CALCULATEURS SIMD

André SEZNEC

Septembre 1984

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE
ET SYSTÈMES ALÉATOIRES

Campus Universitaire de Beaulieu
Avenue du Général Leclerc
35042 - RENNES CÉDEX
FRANCE
Tél. : (99) 36.20.00
Télex : UNIRISA 95 0473 F

UN NOUVEAU RESEAU D'INTERCONNEXION ADAPTE AUX CALCULATEURS SIMD

André SEZNEC

Publication Interne n°228
Juin 1984
42 pages

RESUME :

Nous définissons un ensemble de familles de permutations que le réseau d'interconnexion d'un calculateur SIMD est appelé à réaliser. Nous présentons ensuite un nouveau réseau d'interconnexion facilement dérivable des réseaux d'interconnexion existant déjà, et enfin nous présentons des algorithmes pour calculer la commande de ce réseau pour les permutations appartenant aux familles que nous avons définies.

ABSTRACT :

We first define a set of permutations families that the interconnection network of a SIMD computer will have to perform. Then we present a new interconnection network easily derived from already existing networks. Finally we present algorithms to control this network for the permutations of our families.

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
(L.A. 227)
UNIVERSITÉ DE RENNES 1 I.N.S.A. DE RENNES

INSTITUT NATIONAL DE RECHERCHE
EN INFORMATIQUE ET EN AUTOMATIQUE
(LABORATOIRE DE RENNES)

INTRODUCTION

L'un des problèmes essentiels à résoudre dans la conception des calculateurs vectoriels SIMD (Single Instruction stream Multiple Data stream) est l'accès parallèle aux données à travers le réseau d'interconnexion (figure 0)

Le réseau d'interconnexion le plus simple à concevoir est la matrice de points de croisement, mais sa complexité en nombre de points de croisement est NM (où N est le nombre d'entrées et M le nombre de sorties). Pour pallier ce défaut, d'autres réseaux de complexité moindre ont été définis, citons par exemple :
le réseau oméga de Lawrie (figure 1) [3], le réseau de Benes (figure 5).

En reprenant l'approche de Lenfant [1], [2], nous définissons des familles de permutations que le réseau d'interconnexion d'un calculateur vectoriel SIMD doit pouvoir réaliser. Nous présentons ensuite un nouveau réseau d'interconnexion à 2^n entrées et 2^n sorties, et des algorithmes simples pour calculer les commandes de ce réseau pour ces permutations.

Notre démarche a été d'adapter le réseau d'interconnexion pour résoudre le problème que nous nous posions et non d'essayer de calculer les commandes d'un réseau pour lequel nous savons qu'il est capable de réaliser les permutations que nous désirons comme par exemple le réseau de Benes.

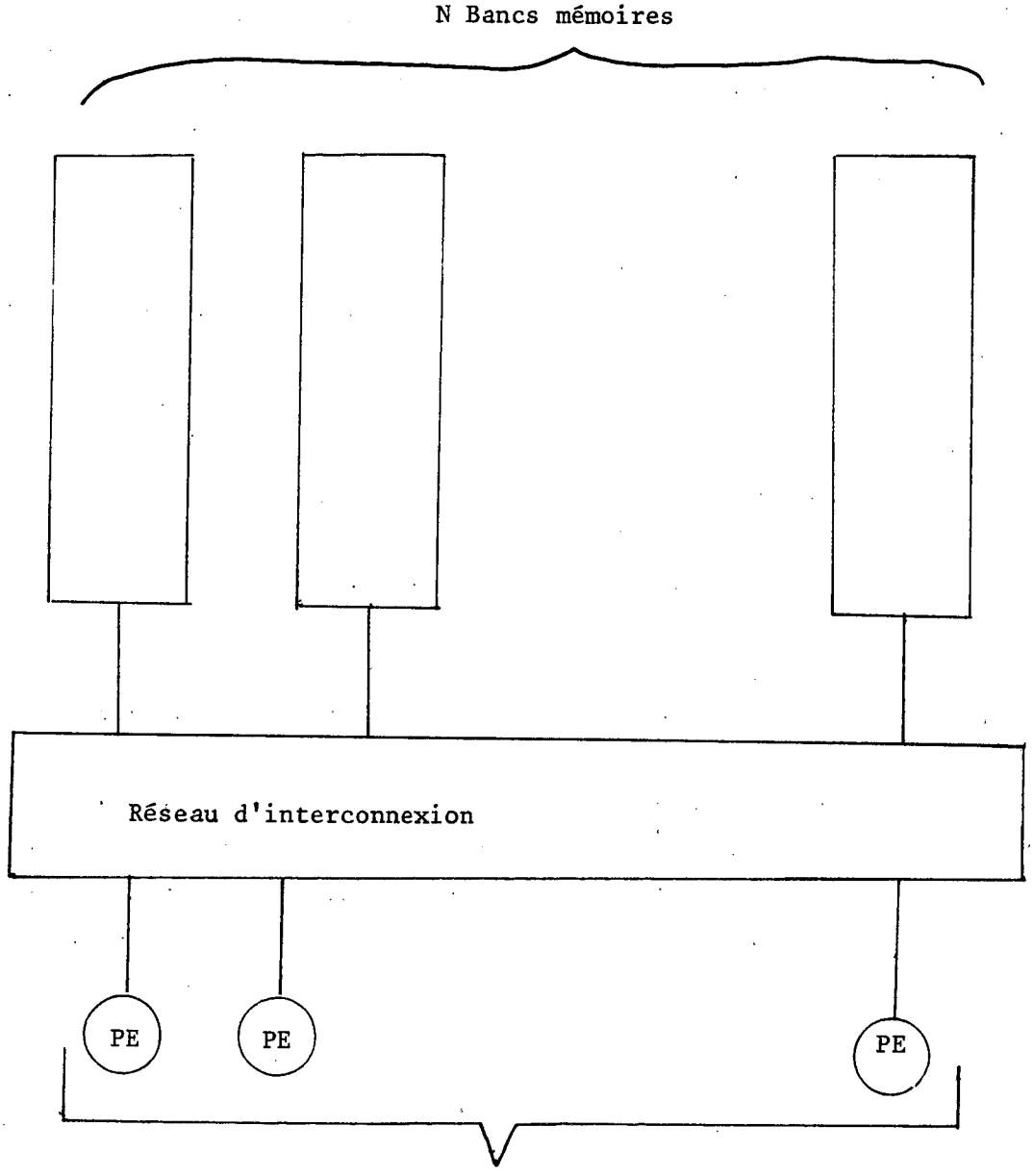


Figure 0 : La structure d'un calculateur SIMD

Table des FUM's de Lenfant

$\sigma^{(n)}$: $x = x_n\, x_{n-1} \cdots x_2\, x_1 \longrightarrow x_{n-1}\, x_{n-2} \cdots x_1\, x_n$

$\rho^{(n)}$: $x = x_n\, x_{n-1} \cdots x_2\, x_1 \longrightarrow x_1\, x_2 \cdots x_{n-1}\, x_n$

$\tau_k^{(n)}$: $x \longrightarrow x \oplus \dot{x}$

$\pi_k^{(n)}$: $x \longrightarrow x + k \bmod 2^n$

Les six familles

$\lambda_{j,k}^{(n)}$: $x \longrightarrow jx + k \bmod 2^n$ (j impair)

$\delta_{j,k}^{(n)}$: $(\hat{a}_2, \hat{a}_1) \longrightarrow (\hat{a}_2, \pi_k^{(n-j)}(\hat{a}_1))$ $(j \leq n)$

$\beta_{j,h,m,k}^{(n)}$: $(\hat{b}_4, \hat{b}_3, \hat{b}_2, \hat{b}_1) \longrightarrow (\hat{b}_4, \hat{b}_1, \hat{b}_2, \hat{b}_3) \oplus k$ $(j + h + m \leq n)$

$\alpha_{u,v,w,k}^{(n)}$: $(\hat{c}_4, \hat{c}_3, \hat{c}_2, \hat{c}_1) \longrightarrow (\hat{c}_4, \rho^{(w-v)}(\hat{c}_3), \hat{c}_2, \rho^{(v)}(\hat{c}_1)) \oplus k$ $(u \leq v \leq w \leq n)$

$\mu_{k,\vec{V}}^{(n)}$: si $v_x = 1$ $x \longrightarrow k + \sum_{y<x} v_y \bmod 2^n$

$\nu_{k,\vec{V}}^{(n)}$: Inverse de $\mu_{k,\vec{V}}^{(n)}$

I PRESENTATION DU PROBLEME

Le concept de vecteur est essentiel en calcul vectoriel. Pour certains, la définition d'un vecteur est la suivante :

"Un vecteur est un ensemble de mots rangés de manière contigüe en mémoire".

Cette définition a le défaut de ne pas être suffisamment souple :

Si une matrice est rangée en mémoire par lignes, une ligne est un vecteur, mais une colonne ou la diagonale ne le sont pas. Pour accéder à une colonne, des réarrangements de données sont donc nécessaires.

Pour pallier ce défaut, nous adopterons la définition suivante :

"Un vecteur est un ensemble ordonné de mots dont les adresses sont en progression arithmétique".

Un vecteur sera donc défini par 2 nombres, l'adresse du premier élément appelée adresse de base et la raison de la progression arithmétique.

Lenfant a sélectionné dans [1] [2] une liste de familles de permutations que le réseau d'interconnexion d'un calculateur SIMD doit être capable de réaliser dans le cas où l'on adopte la première définition du vecteur (Table 1). Il a aussi donné des algorithmes récursifs pour le calcul de la commande d'un réseau de Benes pour les réaliser [1] [2]. Cependant, ces familles de permutations ne sont pas adaptées au cas où un vecteur est défini par adresse de base et raison.

Les transferts de données que nous énumérons ensuite sont parmi les plus importants utilisés en calcul vectoriel.

a) Copie d'un tableau dans un autre

Soient deux tableaux A[0:L] et B[0:L], nous devons être capables de réaliser l'affectation B := A.

Pour pouvoir réaliser cette affectation, dans tous les cas où les deux vecteurs sont représentés avec une raison impaire, le réseau d'interconnexion doit être capable de réaliser la famille

$$\left\{ \lambda_{j,k}^{(n)} \right\}$$

b) Compression d'un tableau A dans un autre B suivant un tableau de booléens C

Soient deux tableaux A[0:L], B[0:J] nous voulons ranger A en B en y supprimant les éléments tels que C(i) = 0.

C'est l'opération COMPRESS de APL ou PACK de FORTRAN 8X.

Cette opération est très utile si nous travaillons sur des vecteurs creux.

Pour pouvoir réaliser cette opération sur des vecteurs représentés avec des raisons impaires, le réseau d'interconnexion doit être capable de réaliser toutes les permutations de la famille

$$\left\{ \lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} \circ \lambda^{(n)}_{p,q} \right\} \quad \text{(voir Lenfant [1])}$$

$\mu^{(n)}_{0,\vec{V}}$ étant une permutation telle que :

$\vec{V}$ étant un vecteur de booléens,

$$\text{si } v_x = 1,\ \mu^{(n)}_{0,V}(x) = \sum_{y<x} v_y$$

Nous préciserons plus loin la permutation que nous allons réaliser. Nous allons montrer une autre utilisation de $\mu^{(n)}_{0,\vec{V}}$.

Nous avons vu plus haut comment réaliser l'affectation B: = A dans le cas où les deux vecteurs sont représentés avec une raison impaire. Supposons maintenant que A est représenté avec une raison paire, $2^k R$ avec R impair et k> 1, et que A a pour adresse de base a.

Soit le vecteur Al $[0:2^k L]$ représenté par l'ensemble des mots rangés aux adresses a+mr avec $m \leq 2^k L-1$

Soit un vecteur de booléens $C[0:2^k L]$ tel que $C(2^k i) = 1$ et $C(2^k i + 1) = 0$ pour $1 \leq i \leq 2^k - 1$

Nous remarquerons immédiatement que $Al(2^k j) = A(j)$

D'où l'affectation B: = A est équivalente à la compression de Al dans B suivant C.

c) Expansion d'un tableau B dans un autre A suivant un tableau de booléens C

C'est l'opération inverse de la compression i.e EXPAND en APL ou UNPACK en FORTRAN 8X.

Les permutations de la famille $\left\{ \lambda^{(n)}_{j,k} \circ \nu^{(n)}_{0,V} \circ \lambda^{(n)}_{p,q} \right\}$ sont alors

requises ($\nu^{(n)}_{0,\vec{V}}$ étant la permutation inverse de $\mu^{(n)}_{0,\vec{V}}$).

d) Copie d'un tableau A dans un autre B après une BP sur A

Soient A [0:L] et B [0:L] deux tableaux de longueur L. Nous supposons $L = 2^p$.

Une permutation M sur $E^{(p)} = \{0,1,\ldots, 2^p-1\}$ est une BP (bit permute permutation) ssi il existe une permutation α sur $\{1,\ldots,p\}$ telle que si $x_p..x_1$ est l'écriture binaire de x alors $M(x) = x_{\alpha^{-1}(p)} \ldots x_{\alpha^{-1}(1)}$

Ces permutations sont très utiles dans de nombreux algorithmes où les vecteurs ont une longueur de la forme 2^p (FFT par exemple).

Exemples de BP : le perfect shuffle $\sigma^{(p)}$

le bit-reversal $\rho^{(p)}$

Il est facile de montrer que pour que le calculateur SIMD soit capable de réaliser ces permutations pour tout p, sur tous les vecteurs A et B rangés avec une raison impaire, le réseau d'interconnexion doit être capable de réaliser la famille

$\left\{\lambda^{(n)}_{j,k} \circ \varphi \circ \lambda^{(n)}_{i,h}\right\}$ où φ est une BPC (bit permute complement permutation) i.e

$\varphi = \tau^{(n)}_d \circ M$ où M est une BP (cf Nassimi et Sahni [7])

Nous avons donc défini les familles de permutations que nous désirons être capables de réaliser sur le réseau d'interconnexion d'un calculateur SIMD. Ce sont des familles déjà connues (cf Lenfant [1] [2], Nassimi et Sahni [7]) que l'on voudrait pré-et post-composer par des éléments de la famille $\left\{\lambda^{(n)}_{j,k}\right\}$.

Lenfant [1] n'a pas pu trouver de commandes simples pour réaliser ces pré-et post-compositions sur le réseau de Benes et suggère donc de réaliser ces permutations en deux voire trois passes sur le réseau de Benes. Notre approche a été de chercher à comprendre pourquoi ces commandes ne sont pas simples à calculer, et ensuite d'adapter le réseau à notre problème.

Nous allons dons tout d'abord justifier le choix du nouveau réseau que nous avons fait.

II PERMUTATIONS ADMISSIBLES TRIANGULAIRES INFERIEURES ET SUPERIEURES

Nous allons étudier deux groupes de permutations de $E^{(n)} = \{0,\ldots, 2^n-1\}$, passant tous les deux sur le réseau omega et sur le réseau oméga renversé.

Ces groupes ont déjà été mis en évidence par Steinberg [4], et nous reprenons ici les mêmes définitions.

Définition 1 Soit une permutation $\pi : E^{(n)} \longrightarrow E^{(n)}$

si $x_n..x_2x_1$ est la représentation binaire de x
si $y_n..y_2y_1$ est la représentation binaire de y

π est dite admissible triangulaire inférieure (respectivement supérieure) si il existe n fonctions booléennes f_i (respectivement g_i) telles que $y_i = x_i \oplus f_i(x_{i-1},.. ,x_1)$ (respectivement $y_i = x_i \oplus g_i(y_n,.., y_{i+1})$)

Nous noterons $\mathcal{L}^{(n)}$ (respectivement $\mathcal{U}^{(n)}$) l'ensemble des permutations admissibles triangulaires inférieures (respectivement supérieures) sur $E^{(n)}$.

Exemples

a) la famille $\left\{\lambda_{j,k}^{(n)}\right\}$ est incluse dans $\mathcal{L}^{(n)}$. Cette remarque simple justifie à elle seule l'intérêt que nous avons porté aux ensembles $\mathcal{L}^{(n)}$ et $\mathcal{U}^{(n)}$.

b) $\mathcal{L}^{(n)} \cap \mathcal{U}^{(n)} = \left\{ \tau_d^{(n)} \right\}_{d \in E^{(n)}}$

Nous jouerons plus loin sur cette constatation pour calculer la commande des permutations de la forme $\varphi_1 \circ \tau_d^{(n)} \circ \varphi_2$ avec $\varphi_1 \in \mathcal{L}^{(n)}$ et $\varphi_2 \in \mathcal{U}^{(n)}$

c) définition: M est une permutation linéaire si : il existe une matrice inversible sur $\mathbb{Z}/_{2\mathbb{Z}}[m_{ij}]$ telle que si $y = M(x)$

$$\begin{pmatrix} y_1 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{pmatrix} = [m_{ij}] \begin{pmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{pmatrix}$$

Nous appellerons $LIN^{(n)}$ l'ensemble des permutations linéaires.

Alors, si $M \in LIN^{(n)}$, $M \in \mathcal{L}^{(n)}$ ssi la matrice associée à M est triangulaire inférieure.

si $M \in LIN^{(n)}$, $M \in \mathcal{U}^{(n)}$ ssi la matrice associée à M est triangulaire supérieure.

Nous rappelons maintenant quelques propriétés de $\mathcal{L}^{(n)}$ et $\mathcal{U}^{(n)}$ montrées par Steinberg [4]

Propriété 1 : $\mathcal{L}^{(n)}$ et $\mathcal{U}^{(n)}$ sont des groupes

Propriété 2 : Si $\pi \in \mathcal{L}^{(n)} \cup \mathcal{U}^{(n)}$, alors π est réalisable par le réseau oméga (et aussi par le réseau oméga renversé)

Propriété 3 : Soit π une permutation appartenant à $\mathcal{L}^{(n)}$:

$$\text{alors} \begin{cases} \Omega^{(n)} = \Omega^{(n)}{}_{\pi} \\ \Omega_R^{(n)} = {}_{\pi}\Omega_R^{(n)} \end{cases}$$

où $\Omega^{(n)}$ est l'ensemble des permutations passant sur le réseau omega, et $\Omega_R^{(n)}$ est l'ensemble des permutations passant sur le réseau omega renversé.

Propriété 4 : Soit u une permutation appartenant à $\mathcal{U}^{(n)}$

$$\text{alors} \begin{cases} \Omega^{(n)} = u\Omega^{(n)} \\ \Omega_R^{(n)} = \Omega_R^{(n)}{}_{u} \end{cases}$$

Nous résumons les propriétés 3 et 4 en disant que $\mathcal{L}^{(n)}$ est inclus dans l'invariant droit de $\Omega^{(n)}$ et l'invariant gauche de $\Omega_R^{(n)}$, et que $\mathcal{U}^{(n)}$ est inclus dans l'invariant gauche de $\Omega^{(n)}$ et l'invariant droit de $\Omega_R^{(n)}$. Steinberg [4] a de plus démontré que ces inclusions sont en fait des égalités.

Pour nous il est intéressant de constater que :

si $\varphi \in \Omega^{(n)}$ alors $\varphi \circ \lambda_{j,k}^{(n)} \in \Omega^{(n)}$

Mais le point le plus important est le suivant :

si $\varphi = \varphi_1 \circ \varphi_2$ avec $\varphi_1 \in \mathcal{U}^{(n)}$ et $\varphi_2 \in \mathcal{L}^{(n)}$ alors $\varphi \in \Omega^{(n)}$ et les commandes du réseau omega pour réaliser φ sont déduites de manière simple à partir des commandes pour réaliser φ_1 et φ_2.

Propriété 5 : si φ est une permutation de $E^{(n)}$ et que $\varphi = \varphi_2 \circ \varphi_1$ avec $\varphi_2 \in \mathcal{U}^{(n)}$ et $\varphi_1 \in \mathcal{L}^{(n)}$

alors la commande pour réaliser φ sur le réseau omega est obtenue en faisant le ou exclusif commutateur par commutateur des commandes pour réaliser

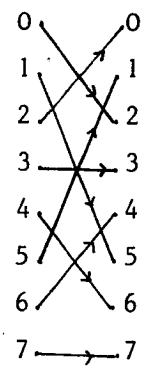
φ_1 et φ_2.

Exemple : n = 3 $\qquad \varphi = U \circ \lambda^{(3)}_{3,2}$

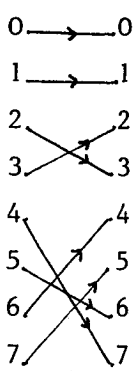
avec $U \in LIN$ la matrice de U étant égale à :

$$M = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

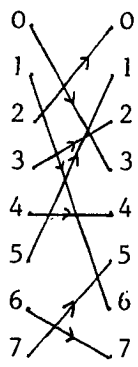
$\lambda^{(3)}_{3,2}$:



U :



φ :



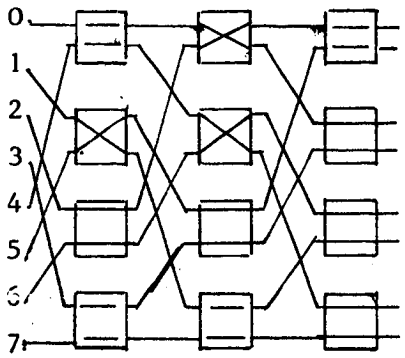


fig. 1 $\lambda_{3,2}^{(3)}$

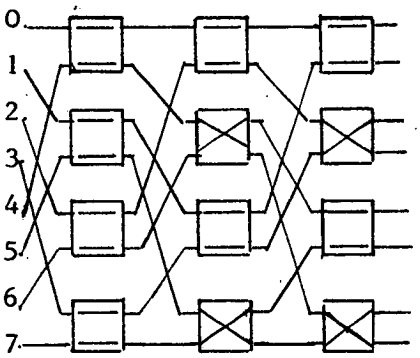


fig. 2 U

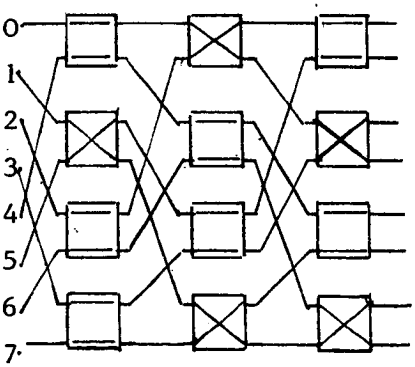


fig. 3 φ (obtenue en faisant le ou exclusif des commandes pour U et $\lambda_{3,2}^{(2)}$)

Démontrons la propriété :

Soit $\varphi_1 \in \mathcal{L}^{(n)}$, alors il existe n fonctions booléennes f_i telles que si $y_n \dots y_1 = \varphi_1(x_n \dots x_1)$ alors $y_i = x_i \oplus f_i(x_{i-1},..,x_1)$

Soit $\varphi_2 \in \mathcal{U}^{(n)}$, alors il existe n fonctions booléennes g_i telles que si $z_n \dots z_i = \varphi_2(y_n,\dots y_1)$ $z_i = y_i \oplus g_i(z_n,..,z_{i+1})$

d'où si $z_n .. z_1 = \varphi_2 \circ \varphi_1(x_n..x_1)$

alors $z_i = x_i \oplus f_i(x_{i-1},..,x_1) \oplus g_i(z_n,..,z_{i+1})$

Rappelons que si $\pi \in \Omega^{(n)}$, alors il existe n fonctions booléennes h_i telles que si $b_n .. b_1 = \pi(a_n..a_1)$ alors $b_i = a_i \oplus h_i(b_n..,b_{i+1},a_{i-1}\dots a_1)$ et que dans ce cas le commutateur ayant pour entrées les éléments d'indices $b_n..b_{i+1}0\ a_{i+1}..a_1$ et $b_n\dots b_{i+1}1\ b_{i-1}..a_1$ est dans l'état $h_i(b_n,..,b_{i+1},a_{i-1},..a_1)$ (voir figure 4)

Dans les cas qui nous intéressent :
Pour réaliser φ_1 le commutateur ayant pour entrées les éléments d'indices $z_n.. z_{i+1}0\ x_{i-1}$ et $z_n.. z_{i+1}1\ x_{i-1}..x$ est donc dans l'état $f_i(x_{i-1},..x)$

Pour réaliser φ_2 il est dans l'état $g_i(z_n,..,z_{i+1})$

Pour réaliser $\varphi_2 \circ \varphi_1$ il est dans l'état $g_i(z_n,..,z_{i+1}) \oplus f_i(x_{i-1},..,x_1)$

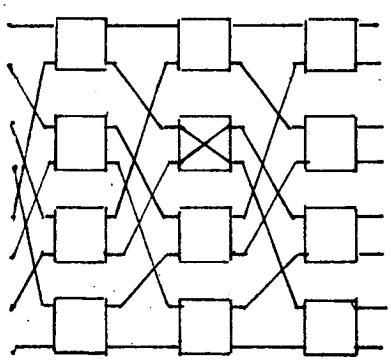


fig.4

Le commutateur marqué a pour entrée 100 et 110 et est dans l'état $h_2(1,0) = 1$

La commande pour réaliser φ_2 o φ_1 sur le réseau omega est donc obtenue en faisant le ou exclusif commutateur par commutateur des commandes pour réaliser φ_1 et φ_2. c.q.f.d.

D'autre part, nous avons la propriété analogue suivante :

Propriété 6 : Soit φ une permutation de $E^{(n)}$ telle que :

$$\varphi = \varphi_1 \text{ o } \varphi_2 \quad \text{avec} \begin{cases} \varphi_1 \in \mathcal{U}^{(n)} \\ \varphi_2 \in \mathcal{L}^{(n)} \end{cases}$$

La commande pour réaliser φ sur le réseau omega est obtenue en faisant le ou exclusif commutateur par commutateur des commandes pour réaliser φ_1 et φ_2.

Ces deux dernières propriétés vont nous être très utiles par la suite.

III LE RESEAU $\mathcal{R}^{(n)}$

Nous rappelons que le réseau de Benes (fig. 5) est équivalent à un réseau omega renversé suivi d'un réseau omega. Ceci et le II entrainent la remarque suivante, il est certainement plus facile de calculer les commandes du réseau de Benes pour des permutations de la forme : u_1 o φ o u_2 avec u_1, u_2 appartenant à $\mathcal{U}^{(n)}$ et φ une permutation dont les commandes sont connues, que celles de l_1 o φ o l_2 avec l_1 et l_2 appartenant à $\mathcal{L}^{(n)}$.

Comme la famille $\left\{\lambda_{j,k}^{(n)}\right\}$ est incluse dans $\mathcal{L}^{(n)}$ et que nous voulons savoir pré-et-post-composer par des éléments de cette famille les éléments des familles $\left\{\mu_{0,v}^{(n)}\right\}$ et $\left\{\nu_{0,v}^{(n)}\right\}$ de Lenfant [1] et BPC de Sahni et Nassimi [7], il est donc naturel de choisir un réseau $\mathcal{R}^{(n)}$ qui soit équivalent à un réseau omega suivi d'un réseau omega renversé.

Nous allons donner trois définitions équivalentes du réseau $\mathcal{R}^{(n)}$

Définition 1 : Le réseau $\mathcal{R}^{(n)}$ est constitué d'un réseau de Benes $\mathcal{B}^{(n)}$ précédé et suivi de la connection $\rho^{(n)}$ (bit-reversal) (fig 5)

Cette définition est donnée pour montrer qu'il est simple de déduire le réseau $\mathcal{R}^{(n)}$ du réseau de Benes $\mathcal{B}^{(n)}$. De cette définition nous déduisons aussi que le réseau $\mathcal{R}^{(n)}$ est capable de réaliser toutes les permutations sur $E^{(n)}$.

Définition 2 : Le réseau $\mathcal{R}^{(n)}$ est constitué d'un réseau omega $\Omega^{(n)}$ suivi d'un réseau omega renversé $\Omega_R^{(n)}$. (Nous pouvons confondre le dernier étage du réseau omega avec le premier étage du réseau omega renversé). (fig. 6)

Cette définition séparant les deux parties de $\mathcal{R}^{(n)}$ nous sera utile pour calculer les commandes dans certains cas.

Dans d'autres cas nous utiliserons une définition récursive du réseau $\mathcal{R}^{(n)}$ analogue à la structure récursive du réseau de Benes.

Définition 3 : (fig 7 et 8)

Nous définissons le réseau $\mathcal{R}^{(n)}$ par récurrence de la manière suivante :

- $\mathcal{R}^{(1)}$ est un simple commutateur
- $n>1$ $\mathcal{R}^{(n)}$ est composé de 3 étages :
  - . Un premier étage de 2^{n-1} commutateurs numérotés de 0 à $2^{n-1}-1$ ayant à leurs entrées les éléments x et $x \oplus 2^{n-1}$
  - . le deuxième étage est constitué de 2 réseaux $\mathcal{R}^{(n-1)}$ que nous appelons $\mathcal{R}_u^{(n-1)}$ et $\mathcal{R}_1^{(n-1)}$. $\mathcal{R}_u^{(n-1)}$ admettant comme entrées les $2^{(n-1)}$ sorties du haut des commutateurs du premier étage, $\mathcal{R}_1^{(n-1)}$ celles du bas.
  - . le dernier étage est constitué de 2^{n-1} commutateurs numérotés de 0 à 2^{n-1} ayant à leurs entrées, les éléments de sorties de $\mathcal{R}_u^{(n-1)}$ et $\mathcal{R}_1^{(n-1)}$ correspondant à leur numéro x et à leurs sorties, les éléments x et $x \oplus 2^{n-1}$

Nous allons maintenant montrer comment calculer la commande de ce réseau pour certaines permutations particulièremant intéressantes.

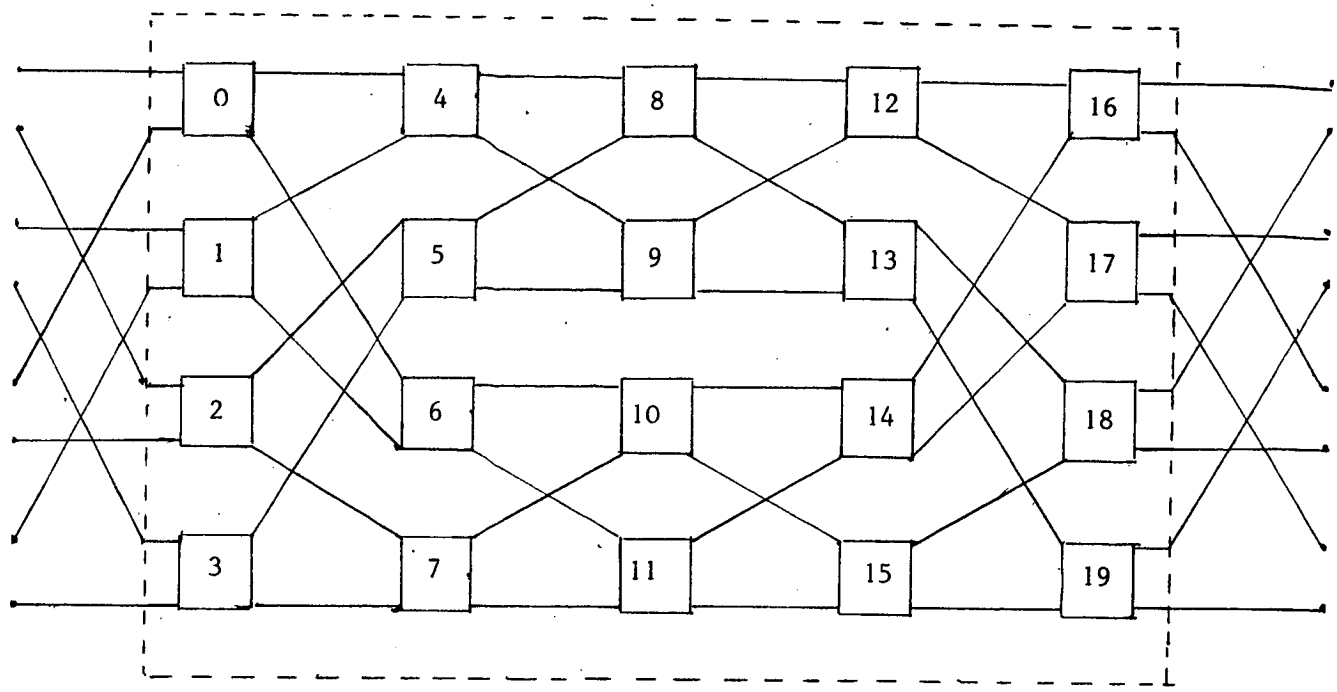


Fig 5 : $\mathcal{R}^{(3)}$ vu comme $\rho^{(3)}\mathcal{B}^{(3)}\rho^{(3)}$

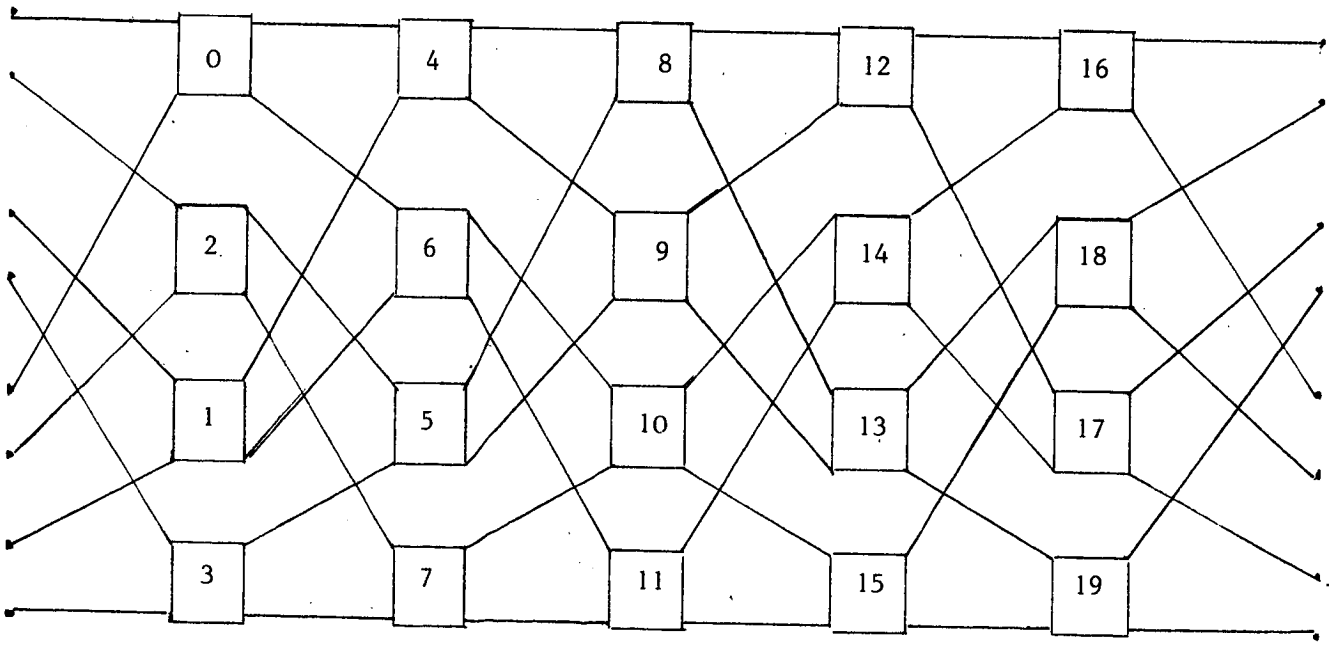


Fig 6 : $\mathcal{R}^{(3)}$ vu comme un réseau $\Omega^{(3)}$ suivi d'un réseau $\Omega_R^{(3)}$ (avec identification du dernier étage du réseau $\Omega^{(3)}$ et du premier étage du réseau $\Omega_R^{(3)}$)

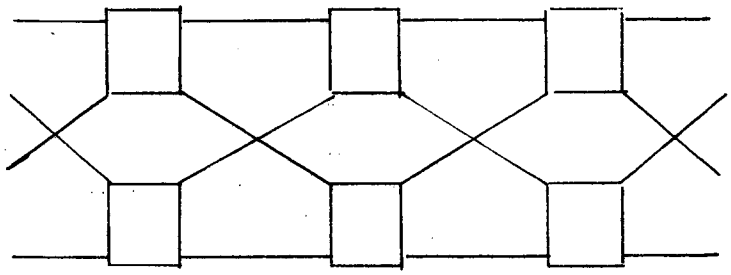


Fig 7 : $\mathcal{R}^{(2)}$ vu de manière récursive

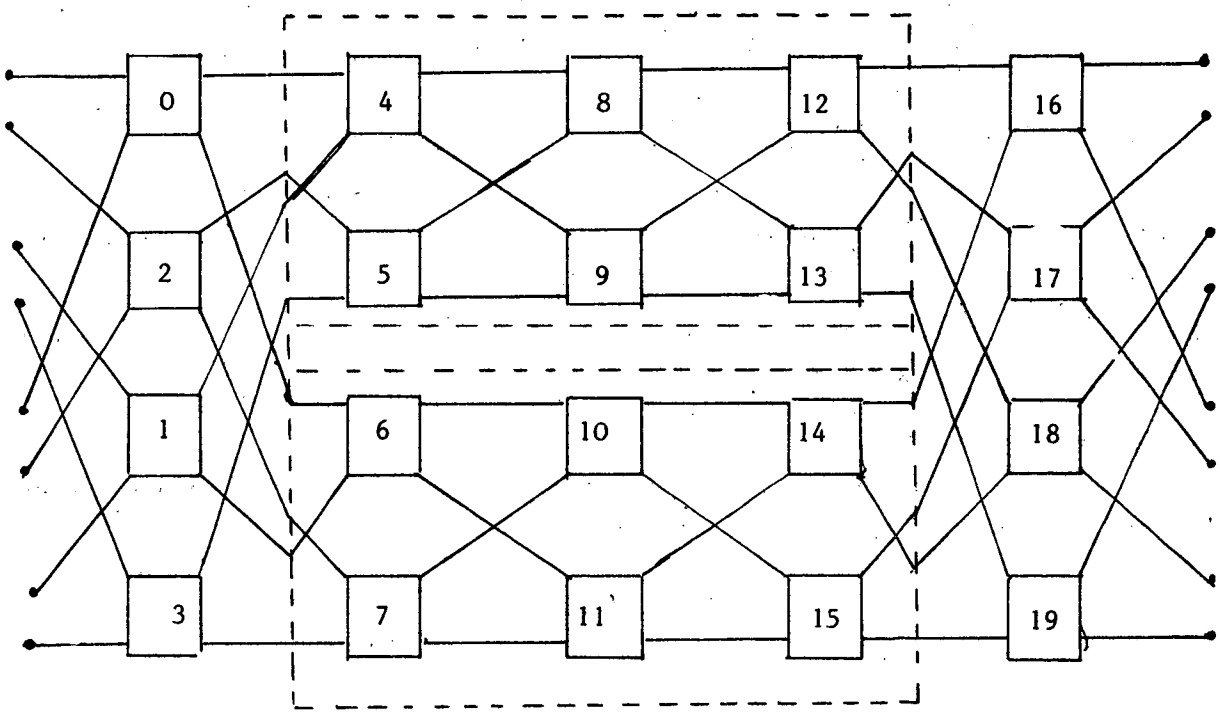


Fig 8 : $\mathcal{R}^{(3)}$ vu de manière récursive

IV CALCUL DE LA COMMANDE DE $\mathcal{R}^{(n)}$ POUR REALISER UNE PERMUTATION AFFINE

Définition : une permutation affine φ est la composée d'une permutation linéaire $M^{(n)}$ d'un ou exclusif $\tau_d^{(n)}$ i.e $\varphi = \tau_d^{(n)} \circ M^{(n)}$ avec $M^{(n)} \in LIN^{(n)}$

Notre but est de calculer la commande de $\lambda_{p,q}^{(n)} \circ \pi \circ \lambda_{z,k}^{(n)}$ sur le réseau $\mathcal{R}^{(n)}$ quand π est une BPC.

Pour cela nous allons montrer comment calculer la commande du réseau $\mathcal{R}^{(n)}$ pour réaliser une permutation affine et en déduire ensuite une décomposition de π sous la forme $L_1 \circ U_1 \circ L_2$ avec L_1, $L_2 \in \mathcal{L}^{(n)}$ et $U_1 \in \mathcal{U}^{(n)}$ pour $\pi \in$ BPC.

D'après les propriétés 5) et 6) il nous suffira donc ensuite de calculer la commande de $L_2 \circ \lambda_{z,k}^{(n)}$ sur le réseau omega, puis celle de $\lambda_{p,q}^{(n)} \circ L_1$ sur le réseau omega renversé, et celle de U_1 sur l'un ou l'autre des deux réseaux.

Par analogie aux notations de Lenfant [1], nous donnerons la commande du réseau $\mathcal{R}^{(n)}$ de manière récursive (cette manière d'envisager la commande est due à la 3ème définition du réseau) :

si φ est une permutation sur $E^{(n)}$ avec $n \geq 2$, nous écrirons la permutation φ sous la forme :

$$\varphi = [\varepsilon_g^{(n)}; (\varphi_0, \varphi_1); \varepsilon_h^{(n)}]$$

$\varepsilon_g^{(n)}$ étant la permutation réalisée par le premier étage du réseau $\mathcal{R}_n$ quand le vecteur de commande est g

φ_0 étant la permutation réalisée par $\mathcal{R}_u^{(n-1)}$

φ_1 étant la permutation réalisée par $\mathcal{R}_l^{(n-1)}$

$\varepsilon_h^{(n)}$ étant la permutation réalisée par le dernier étage du réseau $\mathcal{R}^{(n)}$ quand le vecteur de commande est h.

(Voir fig. 9)

Nous pourrons remarquer que la décomposition de φ n'est jamais unique : si $\varphi = [\varepsilon_g^{(n)}; (\varphi_0, \varphi_1); \varepsilon_h^{(n)}]$ alors $\varphi = [\varepsilon_{\bar{g}}^{(n)}; (\varphi_1, \varphi_0); \varepsilon_{\bar{h}}^{(n)}]$

$\bar{g}$ et $\bar{h}$ étant les vecteurs complémentaires de g et h.

D'autre part si $\varphi = [\varepsilon_g^{(n)}; (\varphi_0, \varphi_1); \varepsilon_h^{(n)}]$ alors $\varphi^{-1} = [\varepsilon_{\bar{h}}^{(n)}; (\varphi_0^{-1}, \varphi_1^{-1}); \varepsilon_{\bar{g}}^{(n)}]$

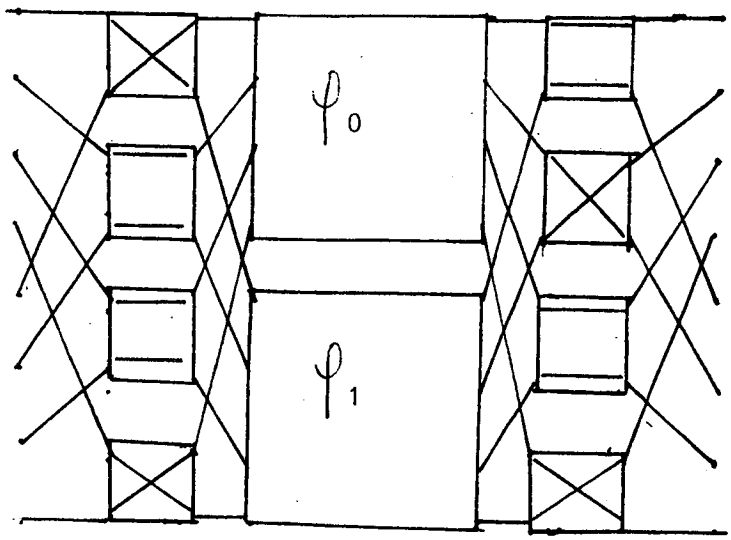


figure 9 . La commande de $\mathfrak{R}^{(3)}$ vue de manière récursive

$$\varphi = \left[\varepsilon^{(3)}_{g} ; (\varphi_0, \varphi_1), \varepsilon^{(3)}_{h}\right] \qquad \text{ici } g = \begin{pmatrix}1\\0\\0\\1\end{pmatrix} \text{ et } h = \begin{pmatrix}0\\1\\0\\1\end{pmatrix}$$

Nous allons donc calculer de manière récursive les commandes pour réaliser les permutations affines sur $\mathcal{R}^{(n)}$

Pour simplifier les notations, nous noterons désormais de la même manière une permutation linéaire et sa matrice associée.

Soit une permutation affine définie par $\varphi = \tau_d^{(n)} \circ M^{(n)}$

la commande pour réaliser φ nous est donnée par le théorème suivant :

Théorème 1

Si $n \geq 2$

posons $d = d' + 2^{n-1} d_n$, $B' = \left(\sum_{c=1}^{n-1} 2^{i-1} M_{i,n}^{(n)}\right) \oplus d'$

soit $M^{(n-1)}$ la matrice $(n-1)\times(n-1)$ définie par :

$$M_{i,j}^{(n-1)} = M_{i,j}^{(n)} \oplus M_{in} . M_{nj}^{(n)}$$

soit ${}_kM^{(n-1)}$ la matrice $(n-1)\times(n-1)$ définie par

$${}_kM_{i,j}^{(n-1)} = M_{ij}^{(n)} \oplus M_{in}^{(n)} \left(M_{nj}^{(n)} \oplus M_{kj}^{(n)}\right)$$

alors a) si $M_{n,n}^{(n)} = 1$ alors

$M^{(n-1)}$ définit une permutation affine.

et $\tau_d^{(n)} \circ M^{(n)} = \left[\varepsilon_g^{(n)} ; \left(\tau_{d'}^{(n-1)} \circ M^{(n-1)} ; \tau_{B'}^{(n-1)} \circ M^{(n-1)}\right) ; \tau_{2^{n-1}d_n}^{(n)}\right]$

avec $g_p = \bigoplus_{i=1}^{n-1} M_{ni}^{(n)} p_i$ si $p = p_{n-1} \cdots p_1$

b) si $M_{n,n}^{(n)} = 0$ alors il existe k tel que $M_{k,n}^{(n)} = 1$ pour $k < n$

et $\varphi = \left[\varepsilon_h^{(n)} ; \left(\tau_{d'}^{(n-1)} \circ_k M^{(n-1)} ; \tau_{B'}^{(n-1)} \circ_k M^{(n-1)}\right) ; \tau_{2^{n-1}(d_n \oplus d_k)}^{(n)} \circ \eta_k^{(n)}\right]$

avec $h_p = \bigoplus_{i=1}^{n-1} (M_{ni}^{(n)} \oplus M_{ki}^{(n)}) p_i$ ou $p = p_{n-1} \cdots p_1$

et $\eta_k^{(n)} (x_n \cdots x_1) = x_n \oplus x_k \; x_{n-1} \cdots x_1$

si $n = 1$ $\varphi = \tau_{d_1}^{(1)}$

Démonstration

a) si $M^{(n)}_{n,n} = 1$

Nous vérifions dans ce cas que $M^{(n-1)}$ est une permutation sur $E^{(n-1)}$ montrons que φ est bien réalisée par le réseau commandé selon le théorème :

si $\varphi(x_n \ldots x_1) = y_n \ldots y_1$

Nous vérifions que $y_n = \bigoplus_{i=1}^{n} M^{(n)}_{n,i}\ x_i \oplus d_n$ comme réalisé par le réseau.

si $\bigoplus_{i=1}^{n} M^{(n)}_{ni}\ x_i = 0$ nous devons vérifier que $\tau^{(n-1)}_{d'} \circ M^{(n-1)}$ transforme

bien $x_{n-1} .. x_1$ en $y_{n-1} .. y_1$

nous avons $M^{(n)}_{n,n} = 1$ donc $x_n = \bigoplus_{j=1}^{n-1} M^{(n)}_{nj}\ x_j$

et nous voulons que $y_i = \left(\bigoplus_{j=1}^{n} M^{(n)}_{i,j} \right) \oplus d_i$ pour $i < n$

or ce que le réseau réalise nous donne :

$$y_i = \left(\bigoplus_{j=1}^{n-1} M^{(n-1)}_{i,j}\ x_j \right) \oplus d_i = \left(\bigoplus_{j=1}^{n-1} \left(M^{(n)}_{ij} \oplus M^{(n)}_{i,n}\ M^{(n)}_{n,j} \right) x_j \right) \oplus d_i$$

$$= \left(\bigoplus_{j=1}^{n-1} M^{(n)}_{ij}\ x_j \right) \oplus M^{(n)}_{i,n} \left(\bigoplus_{j=1}^{n-1} M^{(n)}_{n,j}\ x_j \right) \oplus d_i$$

or $x_n = \bigoplus_{j=1}^{n-1} M^{(n)}_{n,j}\ x_j$

d'où $y_i = \left(\bigoplus_{j=1}^{n-1} M^{(n)}_{ij}\ x_j \right) \oplus d_i$

soit le résultat désiré

De même nous montrons que si :

$$\bigoplus_{i=1}^{n} M^{(n)}_{n,i}\, x_i = 1$$

le réseau réalise le résultat désiré.

b) si $M^{(n)}_{n,n} = 0$

alors comme $M^{(n)}$ est une permutation lineaire, il existe k tel que $M^{(n)}_{k,n} = 1$ et nous vérifions que $\varepsilon^{(n)}_h$ est bien une permutation et qu'elle peut être réalisée par le 1^{er} étage du réseau

soit $T^{(n)}_k$ la matrice définie par $T^{(n)}_{k,i,j} = \delta_{i,j} + \delta_{k,n} \cdot \delta_{j,k}$

soit $M'^{(n)} = T^{(n)}_k M^{(n)}$. Alors comme les matrices sont définies sur $\mathbb{Z}/2\mathbb{Z}$, $T^{(n)-1}_k = T^{(n)}_k$

d'où $M^{(n)} = T^{(n)}_k M'^{(n)}$ et $M'^{(n)}_{n,n} = 1$

il est clair que $T^{(n)}_k$ peut être réalisée par le dernier étage du réseau.

de plus : $\varphi = \tau^{(n)}_d \circ T^{(n)}_k \circ M'^{(n)} = T^{(n)}_k \circ \tau^{(n)}_{d \oplus 2^{n-1} d_k} \circ M'^{(n)}$

en appliquant la partie a) du théorème à $\tau^{(n)}_{d \oplus 2^{n-1} d_k} \circ M'^{(n)}$

Nous obtenons :

$$\tau^{(n)}_{d \oplus 2^{n-1} d_k} \circ M'^{(n)} = \left[\varepsilon^{(n)}_h ; \left(\tau^{(n-1)}_{d'} \circ M'^{(n-1)} ; \tau^{(n-1)}_{d' \oplus B'} \circ M'^{(n-1)} \right) ; \tau^{(n)}_{2^{n-1}(d_n \oplus d_k)} \right]$$

puis par la composition par $T^{(n)}_k$ sur le dernier étage nous avons le résultat souhaité.

cqfd.

Nous pouvons faire quelques remarques :

a) Nous avons décomposé une permutation affine quelconque φ en produit de deux permutations affines $\omega o L$, L étant une permutation de $\mathfrak{L}^{(n)} \cap LIN^{(n)}$ et ω une permutation de $\Omega_R^{(n)}$.

Comme L est aussi réalisable sur le réseau oméga renversé nous pouvons donc énoncer le résultat suivant améliorant un résultat obtenu par Lenfant et Tahé [6] :

Théorème 2 Toute permutation affine peut être réalisée en 2 passes par le réseau oméga renversé.

b) ω est une permutation affine, or Lenfant et Tahé [6] ont montré qu'une permutation affine est réalisable sur le réseau oméga renversé si et seulement si sa matrice est décomposable sous la forme LU, donc toute permutation affine peut s'écrire comme la composition de 3 permutations : $L_1 \circ U_1 \circ L_2$ avec L_2, $L_1 \in \mathfrak{L}^{(n)} \cap LIN^{(n)}$ et $U_1 \in \mathfrak{U}^{(n)}$

c) La matrice de $L^{(n)}$ est facilement obtenue au cours de l'algorithme.

A chaque pas nous obtenons une ligne de $L^{(n)}$:
au premier pas la $n^{ième}$ ligne nous est donnée par $L_{i,n}^{(n)} = M_{i,n}^{(n)}$
pour $i < n$, $L_{n,n}^{(n)} = 1$

Du théorème 1 nous allons déduire le théorème 1 bis.

Théorème 1 bis : $n \geq 2$

Si Ψ est une B P sur $E^{(n)}$, α la permutation associée de $\{1,2,..,n\}$
Si $\tilde{\Psi}$ est la B P sur $E^{(n-1)}$ déduite de Ψ, en posant
$\tilde{\alpha}(\alpha^{-1}(n)) = \alpha(n)$ si $\alpha(n) \neq n$

alors si $\alpha(n) \neq n$

$$\eta^{(n)}_{\alpha} \circ \Psi = \left[\eta^{(n)}_{\alpha^{-1}(n)}; \left(\tau^{(n-1)}_{d'} \circ \tilde{\Psi}, \tau^{(n-1)}_{d' \oplus 2^{\alpha(n)-1}} \circ \tilde{\Psi}\right); \eta^{(n)}_{\alpha(n)} \circ \tau^{(n)}_{2^{n-1}(d_n \oplus d_{\alpha(n)})}\right]$$

Si $\alpha(n) = n$

$$\tau^{(n)}_{d} \circ \Psi = \left[1^{(n)}; \left(\tau^{(n-1)}_{d'} \circ \tilde{\Psi}, \tau^{(n-1)}_{d'} \circ \tilde{\Psi}\right); \tau^{(n)}_{2^{n-1} d_n}\right]$$

Ce théorème nous donne donc la commande de $\mathcal{R}^{(n)}$ pour les éléments de B P C

V Décomposition LUL des permutations appartenant à B P C

Nous allons utiliser le théorème 1 bis pour trouver cette décomposition. soit Ψ une B P, le théorème 1 bis nous donne une décomposition de Ψ sous la forme

$$\Psi = \omega \circ L_1 \quad \text{où } L_1 \in LIN^{(n)} \cap \mathcal{L}^{(n)} \text{ et } \omega \in \Omega_R^{(n)} \cap LIN^{(n)}$$

et nous savons que ω est décomposable sous la forme $L_2 \circ U_2$ avec $U_1 \in \mathcal{U}^{(n)} \cap LIN^{(n)}$ et $L_2 \in \mathcal{L}^{(n)} \cap LIN^{(n)}$

D'où nous avons un algorithme nous donnant Ψ sous la forme $L_1 \circ U_1 \circ L_2$ désirée. Cet algorithme n'est toute fois pas satisfaisant : nous avons à décomposer la matrice de ω sous la forme $L_1 \circ U_1$.

Heureusement nous avons le résultat suivant :

Théorème 3 :

soit Ψ une B P sur $E^{(n)}$

si par l'application du théorème 1 bis, et la décomposition LU à Ψ nous obtenons $\Psi = L_1 \circ U_1 \circ L_2$

et que de la même manière nous obtenons pour Ψ^{-1}

$$\Psi^{-1} = L_1' \circ U_1' \circ L_2'$$

$$\text{alors} \begin{cases} L_1' = L_2^{-1} \\ U_1' = U_1^{-1} \\ L_2' = L_1^{-1} \end{cases}$$

Ce théorème nous donne un moyen pratique de calculer les commandes pour réaliser L_1 sur le réseau omega, L_2 sur le réseau omega renversé, et U_1 sur le réseau omega ou sur le réseau omega renversé :

Nous appliquons le théorème bis à Ψ ($M^{(n)}$ est la matrice associée à Ψ)

nous avons donc la commande de L_2 sur le réseau omega

(il est clair que les commutateurs du dernier étage dont tous à l'état 0)

et la commande de $L_1 \circ U_1$ sur le réseau omega renversé

Puis nous appliquons le théorème à Ψ^{-1} ($^tM^{(n)}$ est associée à Ψ^{-1})

Nous avons alors la commande de L_1^{-1} sur le réseau omega donc celle de L_1 sur le réseau omega renversé et la commande $L_2^{-1} \circ U_1^{-1}$ sur le réseau omega renversé donc celle de $U_1 \circ L_2$ sur le réseau omega

D'après la propriété 1 la commande de U_1 sur le réseau omega est alors obtenue en faisant le ou exclusif de la commande de L_2 avec la commande de $U_1 \circ L_2$. De même la commande de U_1 sur le réseau omega renversé est obtenue grace à la propriété 1.

Dans la pratique nous allons même calculer ainsi la décomposition de $\tau_d^{(n)} \circ \Psi$.

<u>1ère étape</u> : application du théorème 1 bis à $\tau_d^{(n)} \circ \Psi$

ce qui nous donne la commande de $\tau_d^{(n)} \circ L_1 \circ U_1$ sur le réseau omega renversé et la matrice de L_2.

<u>2ème étape</u> : application du théorème 1 bis à Ψ^{-1}

ce qui nous donne la commande de L_1 sur le réseau omega renversé et la matrice de L_1^{-1}

<u>3ème étape</u>: ou exclusif des commandes obtenues et nous obtenons la commande de $\tau^{(n)}_{L_1^{-1}(d)} \circ U_1$ sur le réseau omega renversé

(La commande de $\tau_d^{(n)} \circ L_1 \circ U_1 = L_1 \circ \tau^{(n)}_{L_1^{-1}(d)} \circ U_1$ est obtenue en faisant le ou exclusif des commandes de $\tau^{(n)}_{L_1^{-1}(d)} \circ U_1$ et L_1)

Nous obtenons donc :

$$\tau_d^{(n)} \circ \psi = L_1 \circ (\tau_{L_1^{-1}(d)}^{(n)} \circ U_1) \circ L_2$$

Nous connaissons :

1) la commande de $\tau_{L_1^{-1}(d)}^{(n)} \circ U_1$ sur le réseau omega renversé

2) la matrice de L_2

3) la matrice de L_1^{-1}

Exemple 2

n=3

$$\psi = \sigma^{(3)}$$

et nous voulons connaitre la décomposition de $\tau_5^{(3)} \circ \sigma^{(3)}$

a) par application du théorème 1 à $\tau_5^{(3)} \circ \sigma^{(3)}$

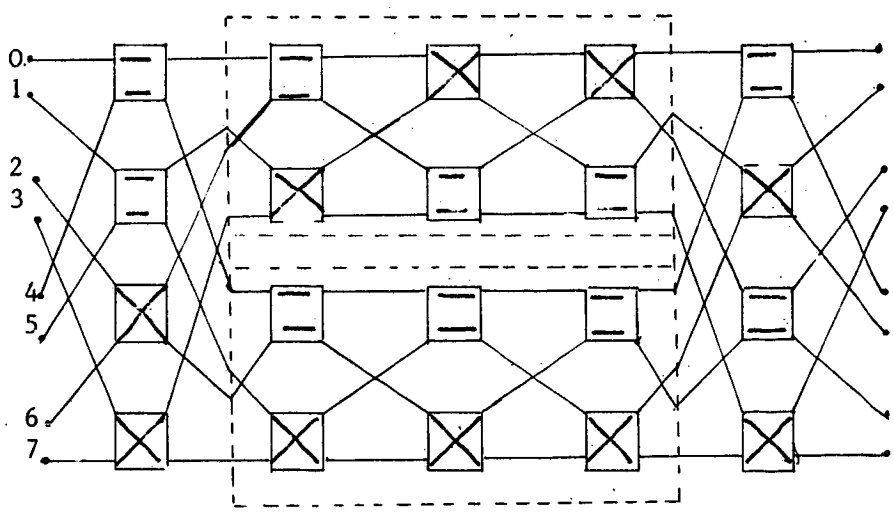


Fig. 10

$$L_2 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

b) par application du théorème 1 à $\sigma^{(3)-1}$ (puis la symétrie pour obtenir $\sigma^{(3)}$)

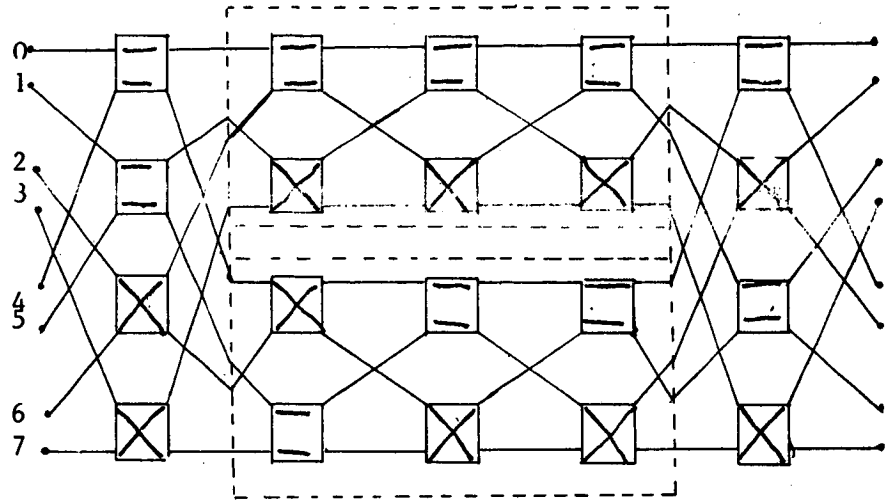


fig. 11

nous obtenons $L_1^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$

D'où la commande du réseau omega renversé pour $\pi^{(3)}_{L_1^{-1}(S)} \circ U_1$

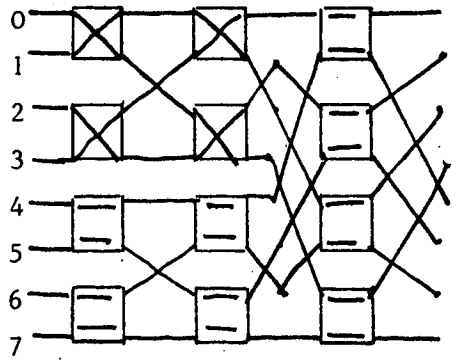


Fig. 12

Démonstration du théorème 3

Appliquons le théorème 1 bis à une, B P Ψ

soit α la permutation associée sur les indices

Nous avons, avec les notations du théorème 1 bis

si $\alpha(n) \neq n$

$$\Psi \equiv \left[\eta^{(n)}_{\alpha(n)} 1_{(n)} \; ; \left(\tilde{\Psi}, \; \tau^{(n-1)}_{2^{\Psi(n)-1}} \circ \tilde{\Psi} \right), \; \eta^{(n)}_{\alpha(n)} \right]$$

si $\alpha(n) = n$

$$\Psi = \left[1^{(n)} ; \; (\tilde{\Psi}, \; \hat{\Psi}), \; 1^{(n)} \right]$$

Il est donc clair si $\alpha(n) = n$ que la décomposition obtenue pour Ψ est la même que pour Ψ^{-1} s'il en est ainsi pour $\tilde{\Psi}$ et $\tilde{\Psi}^{-1}$

si $\alpha(n) \neq n$ nous obtenons

$$\begin{cases} \Psi = \eta^{(n)}_{\alpha(n)} \circ \Psi_1 \circ \eta^{(n)}_{\alpha(n)} & (1) \\ \Psi^{-1} = \eta^{(n)}_{\alpha^{-1}(n)} \circ \Psi_1^{-1} \circ \eta^{(n)}_{\alpha(n)} & (2) \end{cases}$$

avec $\Psi_1 = \left[1^{(n)} \; ; \left(\tilde{\Psi}, \; \tau^{(n-1)}_{2^{\alpha(n)-1}} \circ \tilde{\Psi} \right) ; \; 1^{(n)} \right]$

Ψ_1 a donc pour matrice : $M_{\Psi_1} = \left(\begin{array}{c|c} M_{\tilde{\Psi}} & \\ \hline 00 & 1 \end{array} \right)$

et Ψ_1^{-1} a pour matrice : $M_{\Psi_1^{-1}} = \left(\begin{array}{c|c} M_{\tilde{\Psi}}^{-1} & \\ \hline 00 & 1 \end{array} \right)$

Il est maintenant clair d'aprés (1) et (2) que les décompositions obtenues sont les mêmes pour Ψ et Ψ^{-1} si elles le sont pour $\tilde{\Psi}$ et $\tilde{\Psi}^{-1}$.

Le résultat recherché est donc obtenu par une récurrence immédiate.

c.q.f.d.

VI Commandes pour réaliser $\lambda^{(n)}_{p,q} \circ M \circ \lambda^{(n)}_{z,t}$ ou $M \in \mathcal{BPC}$

Nous avons donc vu au début du IV la marche à suivre. Il nous reste donc à résoudre les problèmes suivants :

- calcul de la commande du réseau oméga pour réaliser $\tau^{(n)}_d \circ L^{(n)} \circ \lambda^{(n)}_{j,k}$ $(L^{(n)} \in LIN^{(n)} \cap \mathcal{U}^{(n)})$
- calcul de la commande du réseau oméga renversé : pour réaliser $\lambda^{(n)}_{j,k} \circ L^{(n)} \circ \tau^{(n)}_d$
- calcul des commandes pour réaliser $U^{(n)} \in LIN^{(n)} \cap \mathcal{U}^{(n)}$ sur le réseau oméga et sur le réseau oméga renversé.

Pour pouvoir donner des formules récursives pour ces commandes nous allons envisager le réseau oméga $\Omega^{(n)}$ de deux manières différentes : nous pouvons considérer le réseau oméga $\Omega^{(n)}$ comme deux réseaux $\Omega^{(n-1)}$ (l'un agissant sur les éléments d'indices pairs et l'autre sur les éléments d'indices impairs) suivi d'un étage qui permet de commuter x et $x \oplus 1$ suivant les valeurs d'un vecteur $\vec{V}$ de booléens (fig. 14)

Cette interprétation nous méne à écrire :

$$\omega = [(\omega_1, \omega_2) ; \varepsilon^{\rightarrow}_{V}]$$

pour ω une permutation passant sur $\Omega^{(n)}$, ω_1, ω_2 étant des permutations passant sur $\Omega^{(n-1)}$ $\varepsilon^{\rightarrow}_{V}$ étant la permutation réalisée par le dernier étage du réseau. $\Omega^{(n)}$ quand le vecteur de commande est égal à $\vec{V}$ de manière duale, nous pouvons considérer le réseau oméga $\Omega^{(n)}$ comme un étage qui permet de commuter x et $x \oplus 2^{n-1}$ suivant les valeurs d'un vecteur $\vec{V}$, suivi de deux réseaux $\Omega^{(n-1)}$ l'un agissant sur les éléments d'indices inférieurs à 2^{n-1}, l'autre sur les éléments d'indices supérieurs à 2^{n-1} (fig. 15)

Cette interprétation nous permet d'écrire :

$$\omega = [\eta^{\rightarrow}_{W} ; (\omega'_1, \omega'_2)]$$

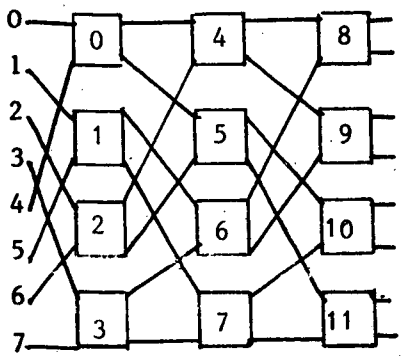


fig. 13 le réseau habituel.

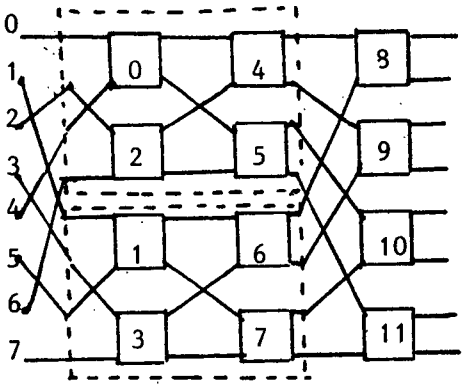


fig. 14 première interprétation

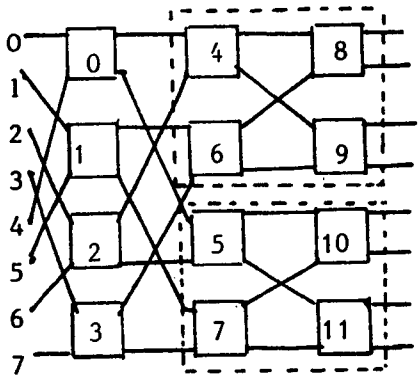


fig. 15 seconde interprétation.

pour ω une permutation passant sur $\Omega^{(n-1)}$ $\eta_{\vec{W}}$ étant la permutation réalisée par le premier étage du réseau oméga $\Omega^{(n)}$ si le vecteur de commande est $\vec{W}$.

Il est clair que la première interprétation est commode pour calculer la commande pour les éléments de $\mathcal{L}^{(n)}_{\vec{V}}$ est le vecteur nul ou son complémentaire, ω_1 et ω_2 appartiennent à $\mathcal{L}^{(n-1)}$. De même la seconde est commode pour calculer la commande pour les éléments de $\mathcal{U}^{(n)}$.

Nous avons aussi des interprétations similaires sur le réseau oméga renversé.

Nous allons maintenant donner les théorèmes permettant le calcul des commandes souhaitées

soit $L^{(n)} \in \mathcal{L}^{(n)} \cap LIN^{(n)}$

soit d un nombre compris entre 0 et 2^n

soit j, k, 2 nombres compris entre 0 et 2^n (j impair)

Nous appellerons

$\xi^{(n)}_{L^{(n)}, d, j, k}$ la permutation définie par :

$$\xi^{(n)}_{L^{(n)}, d, j, k} = \tau^{(n)}_d \circ L^{(n)} \circ \lambda^{(n)}_{j,k}$$

Théorème 4 si on pose $k = 2k' + k_1$, $d = 2d' + d_1$, $j = 2j' + 1$

pour $n \geq 2$, soit $L^{(n-1)}$ la permutation induite par $L^{(n)}$ sur $x_1 = 0$

alors $L^{(n-1)} \in LIN^{(n-1)} \cap \mathcal{L}^{(n-1)}$

si $\begin{cases} L^{(n)}(k_1) = 2a' + k_1 \\ L^{(n)}(k_1 \oplus 1) = 2b' + (k_1 \oplus 1) \end{cases}$

sur le réseau oméga la commande pour $\xi^{(n)}_{L^{(n)}, d, j, k}$ nous est donnée par

$$\xi^{(n)}_{L^{(n)}, d, j, k} = [(\xi^{(n-1)}_{L^{(n-1)}, d' \oplus a', j, k'} \; \xi^{(n-1)}_{L^{(n-1)}, d' \oplus b', j, j'+k'+k_1}); \tau^{(n)}_{k_1 \oplus d_1}]$$

si $n = 1$ $\quad \xi^{(1)}_{L^{(1)}, d, j, k} = \lambda^{(1)}_{k_1 \oplus d_1}$

Démonstration

On note aussi $L^{(n)}$ la matrice associée à $L^{(n)}$

alors si $L^{(n)} = (L_{i,j})$

$A = (a_{ij})$ avec si $j>1$ $a_{i,j} = \delta_{i,j}$

si $j = 1$ $a_{i,1} = L_{i,1}$

$B = (b_{ij})$ avec si $j>1$ $b_{i,j} = L_{i,j}$

et si $j = 1$ $b_{i,1} = \delta_{i,1}$

alors $L^{(n)} = AB$

d'où si on note aussi A et B les permutations linéaires ayant pour matrices A et B

$$\xi_{L^{(n)},\,d,j,k}^{(n)}(x) = \tau_d^{(n)} \circ L^{(n)} \circ \lambda_{j,k}^{(n)}(x)$$
$$= (\tau_d^{(n)} \circ A \circ B \circ \lambda_{j,k}^{(n)})\ (x)$$

si $x = 2x'$

on a alors $B \circ \lambda_{j,k}(x) = 2\,(L^{(n-1)}\ (\lambda_{j,k}^{(n-1)}\ (x'))+k_1$

et $A \circ B \circ \lambda_{j,k}(x) = 2(L^{(n-1)}\ (\lambda_{j,k'}^{(n-1)}\ (x')) \oplus a') + k_1$

$$\tau_d^{(n)} \circ A \circ B \circ \lambda_{j,k}(x) = 2(L^{(n-1)}\ (\lambda_{j,k}^{(n-1)}\ (x')) \oplus a' \oplus d')\ k_1 \oplus d_1$$
$$= 2(\tau_{d'\oplus a'}^{(n-1)} \circ L^{(n-1)} \circ \lambda_{j,k}^{(n-1)}\ (x')) + k_1 \oplus d_1$$

si $x = 2x'+1$

$$P \circ \lambda_{j,k}(x) = 2(L^{(n-1)}\ (\lambda_{j,k'+j'+k_1}^{(n-1)}(x)) + k_1 \oplus 1$$

$$A \circ B \circ \lambda_{j,k}(x) = 2(L^{(n-1)}\,(\lambda_{j,k'+j'+k_1}^{(n-1)}(x)) \oplus b') + k_1 \oplus 1$$

$$\tau_d^{(n)} \circ A \circ B \circ \lambda_{j,k}(x) = 2(\lambda_{d'\oplus b'}^{(n-1)} \circ L^{(n-1)} \circ \lambda_{j,k'+j'+k_1}^{(n-1)}(x)) + k_1 \oplus d_1 \oplus 1$$

Nous avons donc obtenu le resultat cherché

Nouspouvons aussi donner la commande de $\xi_{L^{(n)},d,j,k}^{(n)}$ pour le réseau oméga renversé

<u>Théorème 5</u> sur le réseau oméga renversé

pour $n \geq 2$

$$\xi_{L^{(n)},d,j,k}^{(n)} = \left[\tau_{k_1\oplus d_1}^{(n)} ; \left(\xi_{L^{(n-1)},\,d'\oplus A_0,j,k'+B_0}^{(n-1)} ;\ \xi_{L^{(n-1)},\,d'\oplus A_1,j,k'+B_1}^{(n-1)}\right)\right]$$

avec
$$\begin{cases} A_0 = a'\,\overline{(k_1\oplus d_1)} + b'\ (k_1 \oplus d_1) \\ A_1 = a'\ (k_1\oplus d_1) + b'\ \overline{(k_1\oplus d_1)} \\ B_0 = (j'+k_1)\ (k_1\oplus d_1) \\ B_1 = (j'+k_1)\ \overline{(k_1\oplus d_1)} \end{cases}$$

pour $n = 1$ $\quad \xi_{L^{(n)},d,j,k}^{(n)} = \tau_{k_1\oplus d_1}^{(1)}$

Nous allons maintenant calculer la commande de $\gamma^{(n)}_{L^{(n)},d,j,k} = \lambda^{(n)}_{j,k} \circ (L^{(n)})^{-1} \circ \tau^{(n)}_{d}$

car nous devons nous rappeler que nous connaissons la matrice de $L^{(n)}$ et non de $(L^{(n)})^{-1}$ quand celle-ci est obtenue par le théorème 2 (ou par le théorème 3)

Théorème 6

Sur le réseau oméga renversé la commande pour réaliser $\gamma^{(n)}_{L^{(n)},d,j,k}$ est donnée par :

pour $n \geqslant 2$

$$\text{si} \quad \begin{cases} L^{(n)}(k_1) = 2a' + k_1 \\ L^{(n)}(k_1 \oplus 1) = 2b' + (k_1 \oplus 1) \end{cases}$$

$$\gamma^{(n)}_{L^{(n)},d,j,k} = \left[\tau^{(n)}_{k_1 \oplus d_1} ; \left(\gamma^{(n-1)}_{L^{(n-1)},d' \oplus a', j, k' + (j' + k_1) k_1} \; \gamma^{(n-1)}_{L^{(n-1)},d' \oplus b', k' + (j' + k_1) k_1} \right) \right]$$

$$\text{si } n = 1 \quad \gamma^{(1)}_{L^{(1)},d,j,k} = \tau^{(1)}_{k_1 \oplus d_1}$$

Ce théorème se démontre facilement en constatant que :

$$\gamma^{(n)}_{L^{(n)},d,j,k} = \left(\xi^{(n)}_{L^{(n)},d,j^{-1},-j^{-1},k} \right)^{-1}$$

Nous allons maintenant montrer comment calculer la commande du réseau oméga et du réseau oméga renversé pour realiser $U^{(n)} \in \mathcal{U}^{(n)} \cap LIN^{(n)}$

Théorème 7 Sur le réseau oméga renversé :

soit $U^{(n)} \in \mathcal{U}^{(n)} \cap LIN^{(n)}$

pour $n \geqslant 2$

si $U^{(n)}(2^{n-1}) = 2^{n-1} + e''$

si $d = 2^{n-1} d_n + d''$

si $U^{(n-1)}$ est la restriction de $U^{(n)}$ à $x_n = 0$ alors $U^{(n-1)} \in LIN^{(n-1)} \cap \mathcal{U}^{(n-1)}$

et $\tau^{(n)}_{d} \circ U^{(n)} = \left[\left(\tau^{(n-1)}_{d''} \circ U^{(n-1)}, \tau^{(n-1)}_{d'' \oplus e''} \circ U^{(n-1)} \right) ; \tau^{(n)}_{2^{n-1} d_n} \right]$

si $n = 1 \quad \tau^{(1)}_{d} \circ U^{(1)} = \tau^{(1)}_{d_1}$

Théorème 8 Sur le réseau oméga

pour $n \geq 2$

si $(U^{(n)})^{-1}(2^{n-1}) = 2^{n-1} + f''$

si $d = 2^{n-1}d + d''$

alors $U^{(n)} \circ \tau_d^{(n)} = \left[\tau^{(n)}_{2^{(n-1)}d_n} ; \left(U^{(n-1)} \circ \tau^{(n-1)}_{d''}, U^{(n-1)} \circ \tau^{(n-1)}_{d''+f''} \right) \right]$

si $n = 1 \quad U^{(1)} \circ \tau^{(1)}_d = \tau^{(1)}_{d_1}$

Le théorème 7 sera utilisé quand on connaitra la matrice de $U^{(n)}$, le théorème 8 quand on connaitra la matrice de $U^{(n)-1}$

Nous allons maintenant faire la synthèse des résultats :

Le calcul de la commande de $\lambda^{(n)}_{p,q} \circ \tau^{(n)}_d \circ \Psi \circ \lambda^{(n)}_{j,k}$ sur le réseau $\Omega^{(n)}$ (si Ψ est une E P) se fait en 3 étapes

1) $\begin{cases} \text{calcul de la commande de } \tau^{(n)}_d \circ \Psi \text{ et la matrice de } L_2 \\ \text{calcul de la commande de } \Psi^{-1} \text{ et de la matrice de } L_1^{-1} \end{cases}$

Ces deux calculs peuvent se faire en parallèle. Nous utilisons le théorème 1 bis et le théorème 3

2) - calcul de la commande de $\tau^{(n)}_{L_1(d)} \circ U_1$ sur le réseau oméga renversé (par le ou exclusif de la commande de $\tau^{(n)}_d \circ L_1 \circ U_1$ et de la commande de L_1)

- calcul de la commande de $\lambda^{(n)}_{p,q} \circ L_1$ sur le réseau oméga renversé (par le théorème 6)

- calcul de la commande de $L_2 \circ \lambda^{(n)}_{j,k}$ sur le réseau oméga (par le théorème 4)

Ces trois calculs peuvent se faire en parallèle .

3) Ou exclusif des 2 commandes sur le réseau oméga renversé

Nous pouvons donc admettre que le temps de calcul nécessaire pour calculer est de l'ordre de 2 fois le temps de calcul de la commande d'une FUM de Lenfant [1] sur le réseau de Benes

VII Commandes pour réaliser les permutations des familles $\left\{\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} \circ \lambda^{(n)}_{i,h}\right\}$ et $\left\{\lambda^{(n)}_{j,k} \circ \nu^{(n)}_{0,V} \circ \lambda^{(n)}_{p,q}\right\}$

Nous allons maintenant montrer comment calculer les commandes sur le réseau $\mathcal{R}^{(n)}$ pour ces deux familles liées aux problèmes d'expansion et de compression de vecteurs.

Lenfant [1] a montré que $\nu^{(n)}$ est réalisable par le réseau oméga et que $\mu^{(n)}_{0,V}$ est réalisable par le réseau oméga renversé. Nous allons donc considérer la permutation $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} \circ \lambda^{(n)}_{i,h}$ comme la composition des deux permutations $\lambda^{(n)}_{i,h}$ qui est réalisable par le réseau oméga et $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V}$ qui est réalisable par le réseau oméga renversé.

Pour calculer la commande de $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V}$ sur le réseau oméga renversé, nous allons utilisé un résultat général sur la composition d'une fonction de $\Omega^{(n)}_R$ par une fonction de $\mathcal{L}^{(n)}$ et un résultat obtenu par Lenfant [1]

Lemme 1 : supposons n>2

soit $\varphi \in \mathcal{L}^{(n)}$

sur le réseau oméga $\varphi = [\tau^{(n)}_{k_1} ; (\varphi_0, \varphi_1)]$ avec $\begin{cases} k_1 = 1 \text{ ou } 0 \\ \varphi_0, \varphi_1 \in \mathcal{L}^{(n-1)} \end{cases}$

soit $\Psi \in \Omega^{(n)}_R$

sur le réseau oméga renversé : $\Psi = [\varepsilon^{(n)}_a ; (\Psi_0, \Psi_1)]$

alors sur le réseau oméga renversé la commande de $\varphi \circ \Psi$ nous est donnée par

$$\varphi \circ \Psi = [\tau^{(n)}_{k_1} \circ \varepsilon^{(n)}_a ; (\varphi_0 \circ \Psi_{k_1}, \varphi_1 \circ \Psi_{\bar{k}_1})]$$

La démonstration de ce lemme est immédiate :

si $\varphi = \lambda^{(n)}_{j,k}$ nous avons donc : en posant $k = 2k' + k_1$

$j = 2j' + 1$

$$\lambda^{(n)}_{j,k} \circ \Psi = \left[\tau^{(n)}_{k_1} \circ \varepsilon^{(n)}_a ; \left(\lambda^{(n-1)}_{j,k'+k_1(j'+k_1)} \circ \Psi_{k_1}, \lambda^{(n-1)}_{j,k'+\bar{k}_1(j'+k_1)} \circ \Psi_{\bar{k}_1}\right)\right]$$

Lemme 2 : La commande de $\mu^{(n)}_{0,V}$ sur le réseau oméga renversé nous est donnée par :

$$\text{pour } n\geq 2 \quad \mu^{(n)}_{0,V} = \left[\varepsilon^{(n)}_{g} ; \left(\mu^{(n-1)}_{0,u}, \mu^{(n-1)}_{0,w}\right)\right]$$

$$\text{avec } g_1 = \overline{V_0 \oplus V_1 \cdots \oplus V_{2i}}$$

$$\text{et} \begin{cases} u_1 = V_{2i} \bar{g}_i + V_{2i+1} g_i \\ w_1 = V_{2i+1} \bar{g}_i + V_{2i} g_i \end{cases}$$

$$\text{si } n = 1 \quad \mu^{(1)}_{0,V} = \tau^{(1)}_{\bar{V}_0}$$

(pour la démonstration voir Lenfant [1])

De ces deux lemmes nous allons déduire la commande de $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V}$ sur le réseau oméga renversé :

Théorème 9 Sur le réseau oméga renversé :

$$\text{pour } n\geq 2$$

$$\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} = \left[\varepsilon^{(n)}_{h} ; \left(\lambda^{(n-1)}_{j,k'+k_1(j'+k_1)} \circ \mu^{(n-1)}_{0,\tilde{U}}, \lambda^{(n-1)}_{j,k'+\bar{k}_1(j'+k_1)} \circ \mu^{(n-1)}_{0,\tilde{W}}\right)\right]$$

$$\text{avec } h_i = k_1 \oplus V_0 \oplus \ldots\ldots \oplus V_{2i}$$

$$\text{et} \begin{cases} \tilde{U}_i = V_{2i} \bar{h}_i + V_{2i+1} h_i \\ \tilde{W}_i = V_{2i+1} \bar{h}_i + V_{2i} h_i \end{cases}$$

$$\text{pour } n = 1$$

$$\lambda^{(1)}_{j,k} \circ \mu^{(1)}_{0,V} = \tau^{(1)}_{k_1 \oplus \bar{V}_0}$$

Démonstration

D'aprés les lemmes 1 et 2, nous avons :

$$\text{pour } n\geq 2$$

$$\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} = \left[\tau^{(n)}_{k_1} \circ \varepsilon^{(n)}_{g} ; \left(\lambda^{(n-1)}_{j,k'+k_1(j'+k_1)} \circ \mu^{(n-1)}_{o,\bar{k}_1 u + k_1 w}, \lambda^{(n-1)}_{j,k'+\bar{k}_1(j'+k_1)} \circ \mu^{(n-1)}_{o,k_1 u+\bar{k}_1 w}\right)\right]$$

avec $g_i = \overline{V_o \oplus \ldots . \oplus V_{2i}}$

$$U_{i\ i} = V_{2i}\ \bar{g}_i + V_{2i+1}\ \bar{g}_i$$

$$W_i = V_{2i+1}\ \bar{g}_i + V_{2i}\ g_i$$

Posons $h_i = g_i \oplus k_1 = \overline{k_1 \oplus V_o \ldots . \oplus V_{2i}}$

$$\tilde{U}_i = \bar{k}_1 u_i + k_1 w_i$$

$$\tilde{W}_i = k_1 u_i + \bar{k}_1 w_i$$

Nous allons montrer que :

$$\tilde{U}_i = V_{2i}\ h_i + V_{2i+1} \bar{h}_i$$

Nous avons :

$$\tilde{U}_i = \bar{k}_1\ u_i + k_1\ w_i$$

d'où si $k_1 = 0, \tilde{U}_i = U_i = V_{2i} \bar{g}_i + V_{2i+1} g_i$

or comme $k_1 = 0$, $h_i = g_i$ donc $\tilde{U}_i = V_{2i} \bar{h}_i + V_{2i+1} h_i$

si $k_1 = 1, \tilde{U}_1 = W_i = V_{2i+1} \bar{g}_i + V_{2i} g_i$

or comme $k_1 = 1$, $h_i = \bar{g}_i$ d'où $\tilde{U}_i = V_{2i} h_i + V_{2i+1} h_i$

Donc dans tous les cas :

$$\tilde{U}_i = V_{2i} \bar{h}_i + V_{2i+1} h_i$$

De même nous démontrons que :

$$\tilde{W}_i = V_{2i+1} \bar{h}_i + V_{2i} h_i$$

c.q.f.d.

Rappelons qu'initialement $\mu^{(n)}_{0,V}$ est une permutation telle que :

$$\text{si } V_x = 1, \quad \mu^{(n)}_{0,V}(x) = \sum_{y < x} u_y$$

Il est interessant de noter que nous donnons ici la commande de la permutation telle que :

$$\text{si } V_x = 1 \quad \mu^{(n)}_{0,V}(x) = \sum_{y<x} u_y$$

$$\text{si } V_x = 0 \quad \mu^{(n)}_{0,V}(x) = 2^n - 1 - \sum_{y<x}(1-u_y)$$

Il suffit pour constater ce résultat de noter que :

$$\lambda^{(n)}_{2^n-1,\,2^n-1} \circ \mu^{(n)}_{0,V} = \mu^{(n)}_{0,W} \quad \text{avec } W_i = \bar{V}_i$$

Pour l'opération inverse de la compression, c'est à dire l'expansion, nous avons un résultat analogue :

Théorème 10 Sur le réseau oméga, la commande pour réaliser $\nu^{(n)}_{0,V} \circ \lambda^{(n)}_{j,k}$ nous est donnée par :

si $n \geq 2$

$$\nu^{(n)}_{0,V} \circ \lambda^{(n)}_{j,k} = \left[\left(\nu^{(n-1)}_{0,\tilde{U}} \circ \lambda^{(n-1)}_{j,k'},\ \nu^{(n-1)}_{0,\tilde{W}} \circ \lambda^{(n-1)}_{j,k'+(j'+k_1)}\right);\ \varepsilon^{(n)}_h\right]$$

avec h, $\tilde{U}$, $\tilde{W}$ définis comme pour le théorème 9

si $n = 1$

$$\nu^{(1)}_{0,V} \circ \lambda^{(1)}_{j,k} = \tau^{(1)}_{k_1 \oplus \bar{V}_0}$$

Il est satisfaisant de noter que la complexité du calcul de la commande pour $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} \circ \lambda^{(n)}_{i,h}$ et pour $\lambda^{(n)}_{j,k} \circ \nu^{(n)}_{0,V} \circ \lambda^{(n)}_{i,h}$ sur le réseau $\Omega^{(n)}$ est à peu près la même que celle de la commande pour $\mu^{(n)}_{0,V}$ ou $\nu^{(n)}_{0,V}$ sur le réseau de Benes.

Conclusion

Nous avons présenté le réseau $\mathcal{R}^{(n)}$. Il est facilement dérivable du réseau de Benes $\mathcal{B}^{(n)}$ et ne pose donc pas plus de problèmes de réalisation que celui-ci. Nous avons montré comment calculer les commandes de $\mathcal{R}^{(n)}$ pour les principales permutations que le réseau d'interconnexion d'un calculateur vectoriel SIMD est appelé à réaliser dans le cas où un vecteur est défini par adresse de base et raison.

Pour terminer, nous soulignons le point suivant : en calcul vectoriel les permutations sont plus souvent connues par le nom de leur famille que par les destinations des éléments, donc la commande d'un réseau de la forme matrice de points de croisement n'est pas plus simple que la commande de notre réseau $\mathcal{R}^{(n)}$ au moins pour les familles de permutations que nous avons étudiées (Imaginez le calcul des destinations pour $\lambda^{(n)}_{j,k} \circ \mu^{(n)}_{0,V} \circ \lambda^{(n)}_{i,h}$).

Références :

1- Jacques Lenfant :"A versatile mechanism to move data in an array processor" (à paraitre dans IEEE Transactions on computers)

2- Jacques Lenfant :"Parallel permutations of data : a Benes network control algorithm for frequently used permutations." IEEE Transactions on computers, Vol C 27 PP 635-647, July 1978

3- D. H. Lawrie "Access and alignment of data in a array computer." IEEE Transactions on Computers. Vol. C 24 pp 1145-1155 December 1975

4- D. Steinberg :"Invariant properties of the shuffle-exchange and a simplified cost-effective version of the omega network." IEEE Transactions on Computers, C 32 pp 444-450, May 1983.

5- D. S. Parker :"Notes on Shuffle-exchange-Type Switching Networks." IEEE Transactions on Computers, Vol. C 29, March 1980.

6- Jacques Lenfant and S. Taché : "Permuting data with the omega Network" RADC Final Report Now 1978

7- D. Nassimi and S. Sahni : "An optimal routing algorithm for mesh-connected parallel computers" J. Ass. Comput. Mach. Vol 27 pp 6-29

Imprimé en France
par
l'Institut National de Recherche en Informatique et en Automatique